IN THE CLAIMS:

1. (Cancelled)

2. (Cancelled)

3. (Currently amended) A method for optimizing a loop in an instruction stream of a computer program, the method comprising the steps of:

identifying a partial hot trace using profile data;

identifying an augmented hot trace set of instructions comprising a proper superset of the partial hot trace, but comprising a proper subset of the entire loop, and which forms a complete loop iteration;

unrolling the augmented hot trace set of instructions without unrolling the entire loop;

identifying an augmentation path set using the profile data;

combining the augmentation path set with the partial hot trace to form the augmented hot trace set of instructions; and

~~The method of claim 2,~~ the step of identifying an augmentation path set using the profile data further comprising the step of:

identifying a candidate augmentation path set of instructions having more than one trace through the candidate augmentation path set of instructions, and more than one of the traces having similar probabilities of being executed during an iteration of the loop, the sum of the probabilities of the one or more traces through the set being similar to the probability of execution of the partial hot trace.

4. (Original) The method of claim 3, the step of identifying an augmentation path set further comprising the step of eliminating a particular candidate augmentation path set having a trace having more than a predetermined number of instructions.

5.   (Original) The method of claim 4, wherein the predetermined number of instructions is determined by consideration of system performance.

6.   (Original) The method of claim 4, wherein the predetermined number of instructions is 10.

7.   (Original) The method of claim 4, wherein the predetermined number of instructions is 5.

8.   (Original) The method of claim 3, the step of identifying an augmentation path set further comprises the step of eliminating a particular candidate augmentation path set having more than a predetermined number of traces.

9.   (Original) The method of claim 8, wherein the predetermined number of traces is determined by consideration of system performance.

10.   (Original) The method of claim 8, wherein the predetermined number of traces is 3.

11.   (Original) The method of claim 8, wherein the predetermined number of traces is 2.

12.   (Cancelled)

13.   (Cancelled)

14.   (Currently Amended)   A program product, stored on a computer readable medium, which, when executed by a suitable computer, performs a method of optimizing a loop in an instruction stream of a computer program, the method comprising the steps of:

identifying a partial hot trace using profile data,

identifying an augmented hot trace set of instructions comprising a proper
superset of the partial hot trace, but comprising a proper subset of the entire loop,
and which forms a complete loop iteration;

unrolling the augmented hot trace set of instructions without unrolling the entire
loop;

identifying an augmentation path set using the profile data;

combining the augmentation path set with the partial hot trace to form the
augmented hot trace set of instructions; and

~~The program product of claim 13,~~ the step of identifying an augmentation path set
using <u>the</u> profile data further comprising the step of identifying a candidate
augmentation path set of instructions having more than one trace through the
candidate augmentation path set of instructions, and more than one of the
traces having similar probabilities of being executed during an iteration of the
loop, the sum of the probabilities of the one or more traces through the set
being similar to the probability of execution of the partial hot trace.

15.   (Original) The program product of claim 14, the step of identifying an
augmentation path set using profile data further comprising the step of
eliminating a candidate augmentation path set having a trace with a more than
a predetermined number of instructions.

16.   (Original) The program product of claim 15, wherein the predetermined
number of instructions is determined by consideration of system
performance.

17.   (Original) The program product of claim 15, wherein the predetermined
number of instructions is 10.

18.   (Original) The program product of claim 15, wherein the predetermined
number of instructions is 5.

19.   (Original) The program product of claim 14, the step of identifying an augmentation path set using profile data further comprising the step of eliminating a candidate augmentation path set having more than a predetermined number of traces.

20.   (Original) The program product of claim 19, wherein the predetermined number of instructions is determined by consideration of system performance.

21.   (Original) The program product of claim 19, wherein the predetermined number of traces is 3.

22.   (Original) The program product of claim 19, wherein the predetermined number of traces is 2.